Journal of Critical Reviews
ISSN- 2394-5125 Vol 7, Issue 1, 2020

Data Search using Raita Algorithm

Syihaabul Hudaa[1], Phong Thanh Nguyen[2*], Sri Poedji Lestari[3], G Gunawan[4], S Supiyandi [5]

[1]Institut Teknologi dan Bisnis Ahmad Dahlan Jakarta, Indonesia. Email: hudaasyihaabul@gmail.com
[2]Department of Project Management, Ho Chi Minh City Open University, Vietnam. Email: phong.nt@ou.edu.vn
[3]Universitas Bung Karno, Jakarta, Indonesia
[4]Department of Physics Education, Universitas Mataram, Indonesia
[5]Universitas Pembangunan Panca Budi, Medan, Indonesia

Received: 10.10.2019 Revised: 11.11.2019 Accepted: 12.12.2019

Abstract:
Searching for data is an essential thing to do. The more data stored in a digital archive, the more difficult it is to find that data. Searching requires high speed so that the data is quickly processed. Search requires an excellent algorithm. One of the excellent algorithms in searching data is the Raita algorithm. This algorithm works by comparing search blocks with existing blocks. There ara two processes. The first process is the formation of search blocks and keyword blocks. The second process is to compare what each block is in the BmBc table. This algorithm works well. The application of this algorithm in searching data will be beneficial.

Key Words: Raita, algorithm, searching, data retrieval

 DOI: http://dx.doi.org/10.22159/jcr.07.01.14

INTRODUCTION

One of the common problems handled by computers is to do the search process. The search process was carried out in an organized collection of files or files so that the search process is easy to do. It will not be efficient because the number of archives continues to grow to millions of records. The presence of computer technology makes it easier for humans to search for data because the file is in digital form indexed by a computer, and when the search process is carried out will be able to show in a few seconds because of the search process speed performed by a computer [1]. The search process usually requires an algorithm to speed up the search process; many algorithms can be used for the search process such as Boyer Moore, KMP, Exact String Matching, Brute Force, Raita, Zhu Takaoka, Pattern String Matching and so on, all of these algorithms can be applied in applications for doing the search process [2].

Raita is an algorithm for searching strings or patterns of characters [3]. This algorithm will work by comparing several characters from the middle. These characters will be compared with the characters in the last matching character with (m - 1). The two characters from the back will be compared. The characters in the middle will be compared to other characters, according to the BmBc table. If no one finds a matching character, the search block will shift to the next character. This search will continue until it reaches (n.m + 1). This algorithm is perfect and fast in searching. The results obtained are also very accurate. If there are two similar patterns, this algorithm can detect and report the position of the character to be searched. Raita's algorithm produces a very satisfying output [4].

THEORIES

2.1 Algorithm

Algorithm is come from the Persian mathematician who lived in the 9th century. His name is Abu Abdullah Muhammad ibn Musa Al-Khawarizmi. Initially, the word 'algorithmism' was defined as rules for conducting arithmetic processes using Arabic numerics. The word 'algorithmism' was changed to the word 'algorithm' in the 18th century. Now, the meaning of this word includes all finite procedures for taking out problems or doing work [5]. The initial purpose of the algorithm written for a computer was "There were Byron's notes on the scientific engine," written in 1842, where was considered by most to be the first programmer in the world. Although, since Charles Babbage did not complete his scientific engine, this algorithm was never proved anymore [6].

2.2 Raita Algorithm

Many algorithms can search for strings. One very popular algorithm is the Raita algorithm. This algorithm works quickly in searching string parts in a short or long string. Characters are searched based on patterns and sequences in a parent string. Raita's algorithm works in reverse from the last part of the character. If the searched character is found, Raita's algorithm will search the character from the character in the middle. When searching in the middle of a block, if the character searched has been found, the Raita algorithm will move to other characters. The search will continue from the second character to the two characters of the last character. Then the search will return from starting the middle character.

Raita's algorithm has several stages in carrying out its algorithmic process. Precisely two stages must be done in search. That stage is preprocessing and searching. The following are the steps carried out by Raita's algorithm in searching and matching strings that are applied to two string blocks.

Preprocessing phase

This section in the Raita algorithm aims to calculate the unwell character of the Boyer Moore algorithm. Unwell character is a collection of characters that represent a pattern (pattern). The bad-character shift function of the Boyer Moore algorithm is stored in the bmBc table. To create a bmBc table, the following equation is used.

$$bmBc\ [x[i]\ =\ m - i - 1]$$

Searching phase

The pattern matching process with text starts from the left side of the text. The search process is carried out block by block to find the desired pattern. The process will continue as long as the text is available. If the searched text block has been found, the search will stop.

METHODOLOGY

Analysis and design of the system discussed in the research discuss the process of searching files based on file names that are in the computer by applying the search process using the RAITA algorithm.

The search process using the Raita algorithm has the following steps:
Make the shift table of the pattern sought (P) as the word to be searched for in the text.
If there is a mismatch among the pair of characters at the end of the pattern with the text character, the shift is made from the BmBc table.
If in the process of comparing the final pattern there is a mismatch, the character will be moved again according to the BmBc table
If the end character of the pattern with the characters in the text being compared match, then the position of the character in the pattern and the text will have a value of zero (0), and continue matching at the beginning character of the pattern. If it matches, then it continues matching with the middle character of the pattern.
If the end, beginning and middle of the pattern match. Matching continues with the right side from the beginning of the character in the pattern; if it matches, then it is matched to the center-right of the pattern.
Figure 1 is the Activity Diagram of the system that the author designed to model the system analysis that runs in the file search process using the Raita algorithm. The system will display a search application program with the Raita algorithm. The system will ask whether the program continues or not. The program will request the text to be searched so that the search process can take place. The system will produce text that will be searched whether found or not.

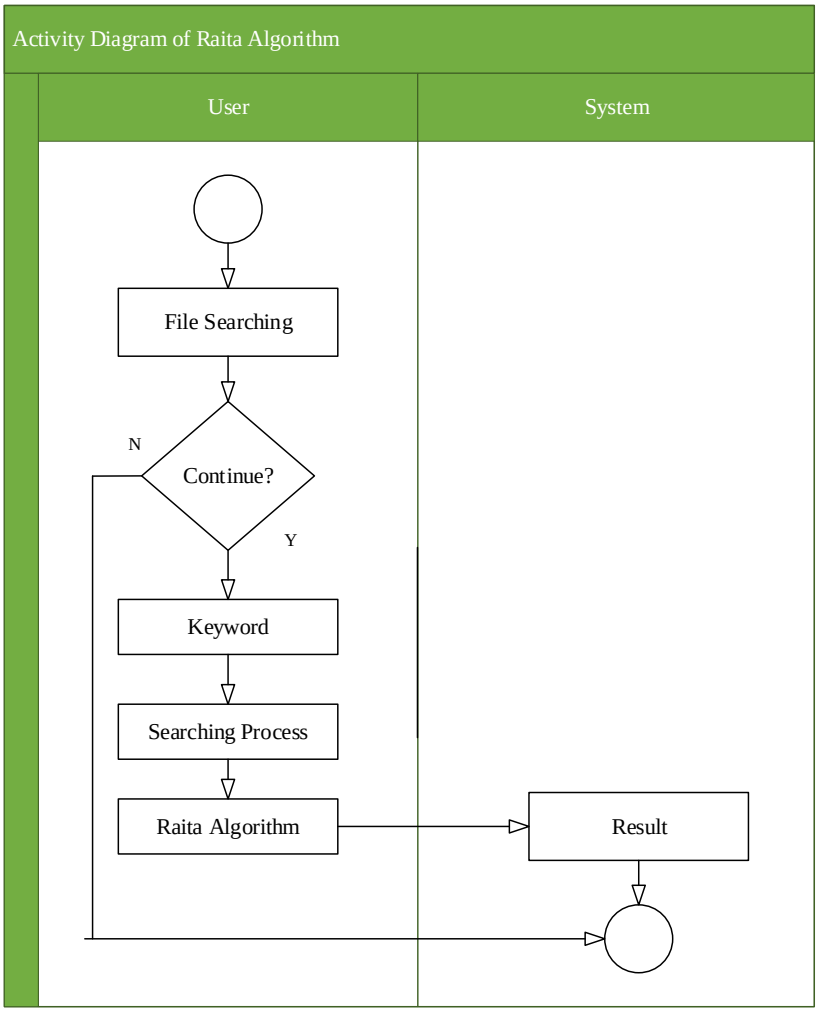

Figure 1. Raita process

RESULT AND DISCUSSION

This section will test the search results. There are two parameters given, P and K. The following is a complete calculation in searching with the parameters given earlier.

P = DOMINOELIZABETH
K = ZABET

Table 1. BmBc

i	Zero	First	Second	Third	Forth
a	Z	A	B	E	T

The following equation is how to create the BmBc table.
$m - 2$
This followin formua shows how the algorithm works:

$m - 1 - i$
$m = K = 5$
Where:

m =
Length of search block
m = P
m = 5-2
= 3
Fomula = m-1-i

5 - 1 - i	(5 - 1 - 0)	4	0	Z
5 - 1 - i	(5 - 1 - 1)	3	1	A
5 - 1 - i	(5 - 1 - 2)	2	2	B
5 - 1 - i	(5 - 1 - 3)	1	3	E
5 - 1 - i	(5 - 1 - 4)	0	4	T

BmBc calculations earlier show the results are as follows:

Table 2. Position

i	Zero	First	Second	Third	Forth	
a	Z	A	B	E	T	*
Result	4	3	2	1	0	5

The pattern length is as many as five characters. Every character that is not found in the BmBC table will be marked "*." This value matches the length of the character block. The Determine the text and sentence patterns to look for, along with the table. search can be seen in stages in the following table by matching character patterns.

Table 3. Text and Sentence Patterns Stage 1

Teks	D	O	M	I	N	O	E	L	I	S	A	B	E	T	H
Pola	Z	A	B	E	T										

In the first stage, there is no match between the letter T and the letter N. The BmBc table is a test table that must be seen. The letter N is not on the table. Because of this case, a pattern change will occur where the pattern will shift by five characters.

The following result is the pattern that is shifted by five characters.

Table 4. Text and Sentence Patterns Stage 2

Teks	D	O	M	I	N	O	E	L	I	Z	A	B	E	T	H
Pola						Z	A	B	E	T					

In this section, the characters that are tested are in accordance with the characters in the BmBc table. The T character will be seen in the Z character table. The pattern will shift by two.

Table 5. Text and Sentence Patterns Stage 3

Teks	D	O	M	I	N	O	E	L	I	Z	A	B	E	T	H
Pola								Z	A	B	E	T			

The next step proves that the T character has the same value in the block. The T value is listed in the BmBc table. The value is zero. Subsequent tests will be performed for each character. Tests start from character 0 to character 4. It matches the position of the character in the table. The search for all the characters tested matches the one in the search block.

Table 6. Text and Sentence Patterns Stage 4

Teks	D	O	M	I	N	O	E	L	I	Z	A	B	E	T	H
Pola										Z	A	B	E	T	

CONCLUSION

Based on research that has been done, several conclusions can be explained. Raita's Algorita is an excellent algorithm for string matching. This algorithm works quickly by comparing block by block in the search linkup space. This algorithm produces accurate data. Raita's algorithm can be used to search text or words from a paragraph. Raita's algorithm can still search text or words consisting of several paragraphs. The speed of the pattern matching process with the text depends on the length of the character pattern used, the longer the character pattern, the faster the matching process and vice versa.

REFERENCES

1. D. G. Kirkpatrick and R. Seidel, "The Ultimate Planar Convex Hull Algorithm?," *SIAM J. Comput.*, vol. 15, no. 1, pp. 287–299, Feb. 1986.
2. R. Anggraini, N. A. Hasibuan, S. Suginam, and F. T. Waruwu, "IMPLEMENTASI ALGORITMA RAITA SEARCH PADA APLIKASI FILSAFAT BERBASIS ANDROID," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 2, no. 1, Oct. 2018.
3. R. Rahim *et al.*, "Searching Process with Raita Algorithm and its Application," *J. Phys. Conf. Ser.*, vol. 1007, no. 1, pp. 1–7, 2018.
4. S. S. Adi *et al.*, "Repetition-free longest common subsequence," *Discret. Appl. Math.*, vol. 158, no. 12, pp. 1315–1324, Jun. 2010.

5. E. R. Firmansyah, "Algoritma Kriptografi & Contohnya," *Univ. Islam Negeri Syarif Hidayatullah Jakarta*, 2012.

6. R. Sedgewick and K. Wayne, *Algorithms*, 4th ed. Addison-Wesley Professiona, 2011.